



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/783,002	02/20/2004	Fuhwei Lwo	RSW920030291US1	7441

23550 7590 04/17/2007  
HOFFMAN WARNICK & D'ALESSANDRO, LLC  
75 STATE STREET  
14TH FLOOR  
ALBANY, NY 12207

EXAMINER
----------

CHEN, QING

ART UNIT	PAPER NUMBER
----------	--------------

2191

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	04/17/2007	PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

# Office Action Summary

Application No.

10/783,002

Applicant(s)

LWO, FUHWEI

Examiner

Qing Chen

Art Unit

2191

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

## Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 20 February 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-27 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-27 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 20 February 2004 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☒ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

## Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

## Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO/SB/08)  
Paper No(s)/Mail Date 20040220.
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_.
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: \_\_\_\_\_.

### DETAILED ACTION

1. This is the initial Office action based on the application filed on February 20, 2004.
2. **Claims 1-27** are pending.

#### *Oath/Declaration*

3. The oath or declaration is defective. A new oath or declaration in compliance with 37 CFR 1.67(a) identifying this application by application number and filing date is required. See MPEP §§ 602.01 and 602.02.

The oath or declaration is defective because:

It does not identify the mailing address of each inventor. A mailing address is an address at which an inventor customarily receives his or her mail and may be either a home or business address. The mailing address should include the ZIP Code designation. If the mailing address of the inventor is the same as the residence address of the inventor, the phrase "Same as the residence" may be used to identify the mailing address. The mailing address may be provided in an application data sheet or a supplemental oath or declaration. See 37 CFR 1.63(c) and 37 CFR 1.76.

#### *Specification*

4. The abstract of the disclosure is objected to because:
  - The abstract must commence on a separate sheet, preferably following the claims, under the heading "Abstract" or "Abstract of the Disclosure." See 37 CFR § 1.72(b).
  - The trademark, JAVA, should be capitalized wherever they appear (capitalize each letter OR accompany each trademark with an appropriate designation symbol, *e.g.*, <sup>TM</sup> or ®) and be accompanied by the generic terminology (use trademarks as adjectives modifying a descriptive noun, *e.g.*, "the JAVA programming language").Correction is required. See MPEP § 608.01(b).

5. The disclosure is objected to because of the following informalities: “jar” should be changed to uppercase on page 11, paragraph [0027]; page 13, paragraph [0030]; and page 14, paragraph [0033].

Appropriate correction is required.

6. The use of trademarks, such as JAVA, JVM, and RATIONAL, has been noted in this application. Trademarks should be capitalized wherever they appear (capitalize each letter OR accompany each trademark with an appropriate designation symbol, *e.g.*, <sup>TM</sup> or ®) and be accompanied by the generic terminology (use trademarks as adjectives modifying a descriptive noun, *e.g.*, “the JAVA programming language”).

Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner, which might adversely affect their validity as trademarks.

#### ***Claim Objections***

7. **Claims 2-9** are objected to because of the following informalities:

- **Claims 2-9** recite the statutory category of invention “The method.” Applicant is advised to change this statutory category of invention to read “The computer-implemented method” for the purpose of providing it with proper explicit antecedent basis.

Appropriate correction is required.

***Claim Rejections - 35 USC § 112***

8. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

9. **Claims 1-27** are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

**Claims 1, 3-6, 9, 10, 12-15, 18, 19, 21-24, and 27** contain the trademark or trade name JAVA. When a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of the 35 U.S.C. 112, second paragraph. *Ex parte Simpson*, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, the use of a trademark or trade name in a claim to identify or describe a material or product (in the present case, a specific programming language) would not only render a claim indefinite, but would also constitute an improper use of the trademark or trade name.

**Claims 2, 7, and 8** depend on Claim 1 and, therefore, suffer the same deficiency as Claim 1.

**Claims 11, 16, and 17** depend on Claim 10 and, therefore, suffer the same deficiency as Claim 10.

**Claims 20, 25, and 26** depend on Claim 19 and, therefore, suffer the same deficiency as Claim 19.

***Claim Rejections - 35 USC § 101***

10. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

11. **Claims 10-18** are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

**Claims 10-18** are directed to systems. However, the recited components of the systems appear to lack the necessary physical components (hardware) to constitute a machine or manufacture under § 101. Therefore, these claim limitations can be reasonably interpreted as computer program modules—software *per se*. Furthermore, the specification discloses that the present invention can be realized in software (*see Page 15, Paragraph [0035]*). The claims are directed to systems of functional descriptive material *per se*, and hence non-statutory.

The claims constitute computer programs representing computer listings *per se*. Such descriptions or expressions of the programs are not physical “things.” They are neither computer components nor statutory processes, as they are not “acts” being performed. Such claimed computer programs do not define any structural and functional interrelationships between the

Art Unit: 2191

computer program and other claimed elements of a computer, which permit the computer program's functionality to be realized. In contrast, a claimed computer-readable medium encoded with a computer program is a computer element, which defines structural and functional interrelationships between the computer program and the rest of the computer, that permits the computer program's functionality to be realized, and is thus statutory. See *Lowry*, 32 F.3d at 1583-84, 32 USPQ2d at 1035.

***Claim Rejections - 35 USC § 103***

12. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

13. **Claims 1-6, 8, 10-15, 17, 19-24, and 26** are rejected under 35 U.S.C. 103(a) as being unpatentable over Schwabe (US 6,986,132) in view of Judge et al. (US 6,430,564).

As per **Claim 1**, Schwabe discloses:

- receiving source input corresponding to a first release of Java™ byte code and target input corresponding to a second release of the Java™ byte code (*see Figure 20A: 1540 and 1550*);
- transforming the source input into a first list that contains Java™ class names associated with the first release of Java™ byte code, and the target input into a second list

Art Unit: 2191

containing Java™ class names associated with the second release of the Java™ byte code (see *Figure 20A: 1535 and 1545; Column 14: 14-15, "An API definition file defines the context of a binary file in relationship to other referenced binary files."*);

- finding matching class names between the first list and the second list, and loading classes corresponding to the matching class names (see *Column 4: 1-10, "In the JVM, the loading step retrieves the class file representing the desired class."*; *Column 25: 44-48, "If the set of classes and interfaces defined in the old API definition file is not found in the new API definition file ..."*); and

- comparing the loaded classes to identify APIs that have been modified between the first release of Java™ byte code and the second release of the Java™ byte code (see *Column 25: 50-53, "... the class and interface attributes are compared to the attributes of the same class or interface in the new package. The attributes may include the name, flags, number of fields and number of methods."*).

However, Schwabe does not disclose:

- removing the matching class names from the first list and the second list after the comparing, wherein any class names remaining in the first list represent APIs that have been removed for the second release of the Java™ byte code, and wherein any class names remaining in the second list represent APIs that have been added for the second release of the Java™ byte code.

Judge et al. disclose:

- removing the matching class names from the first list and the second list after the comparing, wherein any class names remaining in the first list represent APIs that have been



Art Unit: 2191

removed for the second release of the Java™ byte code, and wherein any class names remaining in the second list represent APIs that have been added for the second release of the Java™ byte code (see Column 4: 63-67 through Column 5: 1-5, "Method unloadDataClass unloads a data class by the name of "dataName" by removing the data class object and all instances of the data class object from the data cache 54. Upon removal of a data class object from the data cache 54, the name of the data class "dataName" is also removed from the data class list 47 maintained by Data Manager 48.").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Judge et al. into the teaching of Schwabe to include removing the matching class names from the first list and the second list after the comparing, wherein any class names remaining in the first list represent APIs that have been removed for the second release of the Java™ byte code, and wherein any class names remaining in the second list represent APIs that have been added for the second release of the Java™ byte code. The modification would be obvious because one of ordinary skill in the art would be motivated to determine differences between two data lists.

As per **Claim 2**, the rejection of **Claim 1** is incorporated; and Schwabe further discloses:

- outputting a report identifying at least one of the APIs that have been modified, the APIs that have been removed and the APIs that have been added (see Column 25: 44-67 through Column 26: 1-10, "... a verification error is indicated.").

As per **Claim 3**, the rejection of **Claim 1** is incorporated; and Schwabe further discloses:

Art Unit: 2191

- wherein the loading step comprises loading at least one Java™ class of the first release of Java™ byte code and at least one Java™ class of the second release of the Java™ byte code (*see Column 4: 1-10, "In the JVM, the loading step retrieves the class file representing the desired class."*).

As per **Claim 4**, the rejection of **Claim 3** is incorporated; and Schwabe further discloses:

- listing methods of the at least one Java™ class of the first release of Java™ byte code in the first list, and listing methods of the at least one Java™ class of the second release of the Java™ byte code in the second list (*see Column 26: 6-10, "At 1655, for each method in the old package, the attributes are compared to the same method in the new package. The attributes may include the name, flags and signature."*).

As per **Claim 5**, the rejection of **Claim 4** is incorporated; and Schwabe further discloses:

- wherein the comparing step comprises comparing the methods in the first list to the methods in the second list to identify APIs that have been modified between the first release of Java™ byte code and the second release of the Java™ byte code (*see Column 25: 50-53, "... the class and interface attributes are compared to the attributes of the same class or interface in the new package. The attributes may include the name, flags, number of fields and number of methods."*).

As per **Claim 6**, the rejection of **Claim 5** is incorporated; however, Schwabe does not disclose:

- wherein the removing step comprises removing, from the first list and the second list, any methods in the first list that are identical to methods in the second list based on the comparison, wherein any methods remaining in the first list after the removing represent APIs that have been removed for the second release of the Java™ byte code, and wherein any methods remaining in the second list after the removing represent APIs that have been added for the second release of the Java™ byte code.

Judge et al. disclose:

- wherein the removing step comprises removing, from the first list and the second list, any methods in the first list that are identical to methods in the second list based on the comparison, wherein any methods remaining in the first list after the removing represent APIs that have been removed for the second release of the Java™ byte code, and wherein any methods remaining in the second list after the removing represent APIs that have been added for the second release of the Java™ byte code (*see Column 4: 63-67 through Column 5: 1-5, "Method unloadDataClass unloads a data class by the name of "dataName" by removing the data class object and all instances of the data class object from the data cache 54. Upon removal of a data class object from the data cache 54, the name of the data class "dataName" is also removed from the data class list 47 maintained by Data Manager 48."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Judge et al. into the teaching of Schwabe to include wherein the removing step comprises removing, from the first list and the second list, any methods in the first list that are identical to methods in the second list based on the comparison, wherein any methods remaining in the first list after the removing represent APIs

Art Unit: 2191

that have been removed for the second release of the Java™ byte code, and wherein any methods remaining in the second list after the removing represent APIs that have been added for the second release of the Java™ byte code. The modification would be obvious because one of ordinary skill in the art would be motivated to determine differences between two data lists.

As per **Claim 8**, the rejection of **Claim 1** is incorporated; and Schwabe further discloses:

- wherein the source input and the target input comprise a list of classes (*see Column 22: 56-58, "A library or applet package (herein referred to as a binary file) is received (1460) ... "*).

**Claims 10-15 and 17** are system claims corresponding to the method claims above (Claims 1-6 and 8) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 1-6 and 8.

**Claims 19-24 and 26** are program product claims corresponding to the method claims above (Claims 1-6 and 8) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 1-6 and 8.

14. **Claims 7, 9, 16, 18, 25, and 27** are rejected under 35 U.S.C. 103(a) as being unpatentable over Schwabe (US 6,986,132) in view of Judge et al. (US 6,430,564) as applied to Claims 1, 10, and 19 above, and further in view of Connelly et al. (US 6,385,722).

As per **Claim 7**, the rejection of **Claim 1** is incorporated; however, Schwabe and Judge et al. do not disclose:

- wherein the source input and the target input comprise JAR files.

Connelly et al. disclose:

- wherein the source input and the target input comprise JAR files (*see Column 1: 14-17, "Software vendors typically ship their products as a set of shared libraries, such as libraries written in the Java™ object-oriented programming language and packaged as a conventional shared library file called a JAR file."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Connelly et al. into the teaching of Schwabe to include wherein the source input and the target input comprise JAR files. The modification would be obvious because one of ordinary skill in the art would be motivated to easily and efficiently share and use these library files (*see Connelly et al. – Column 1: 17-20*).

**Claim 16** is rejected for the same reason set forth in the rejection of Claim 7.

**Claim 25** is rejected for the same reason set forth in the rejection of Claim 7.

As per **Claim 9**, the rejection of **Claim 1** is incorporated; however, Schwabe and Judge et al. do not disclose:

- inputting class paths common to the first release of Java™ byte code and the second release of the Java™ byte code.

Connelly et al. disclose:

- inputting class paths common to the first release of Java™ byte code and the second release of the Java™ byte code (*see Column 7: 46-48, "... the class java.net.URLClassLoader is used as class loader 122 to load classes and resources from a class path of JAR files and directory URLs."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Connelly et al. into the teaching of Schwabe to include inputting class paths common to the first release of Java™ byte code and the second release of the Java™ byte code. The modification would be obvious because one of ordinary skill in the art would be motivated to access the parts of shared libraries (*see Connelly et al. – Column 1: 31-34*).

**Claim 18** is rejected for the same reason set forth in the rejection of Claim 9.

**Claim 27** is rejected for the same reason set forth in the rejection of Claim 9.

### ***Conclusion***

15. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM. The Examiner can also be reached on alternate Fridays.

Art Unit: 2191

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Wei Zhen, can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

QC / QC  
April 3, 2007



WEI ZHEN  
SUPERVISORY PATENT EXAMINER